Integrating expert's knowledge constraint of time dependent exposures in structure learning for Bayesian networks

Vahé Asvatourian, Philippe Leray, Stefan Michiels, Emilie Lanoy

Integrating expert's knowledge constraint of time dependent exposures in structure learning for Bayesian networks

Vahé Asvatourian[1, 2], Philippe Leray[3], Stefan Michiels[1,2], Emilie Lanoy[1,2]

[1] Paris-Saclay university, Paris-Sud Univ., UVSQ, CESP, INSERM, Villejuif, France;
[2] Biostatistics and Epidemiology unit, Gustave-Roussy, Villejuif, France;
[3]LS2N UMR 6004, DUKe research group, University of Nantes, France

Submitting author: Vahé Asvatourian,
Gustave Roussy batiment B2M
114 Edouard-Vaillant
94805 Villejuif-cedex
Phone :+33(0)142116508

Email addresses
vahe.asvatourian@gmail.com
philippe.Leray@univ-nantes.fr
Stefan.MICHIELS@gustaveroussy.fr
Emilie.LANOY@gustaveroussy.fr

Abstract

Learning a Bayesian network is a difficult and well known task that has been largely investigated. To reduce the number of candidate graphs to test, some authors proposed to incorporate a priori expert knowledge. Most of the time, this a priori information between variables influences the learning but never contradicts the data. In addition, the development of Bayesian networks integrating time such as dynamic Bayesian networks allows identifying causal graphs in the context of longitudinal data. Moreover, in the context where the number of strongly correlated variables is large (i.e. oncology) and the number of patients low; if a biomarker has a mediated effect on another, the learning algorithm would associate them wrongly and vice versa. In this article we propose a method to use the a priori expert knowledge as hard constraints in a structure learning method for Bayesian networks with a time dependant exposure. Based on a simulation study and an application, where we compared our method to the state of the art PC-algorithm, the results showed a better recovery of the true graphs when integrating hard constraints a priori expert knowledge even for small level of information.

Keywords: Dynamic Bayesian network, graphical structure learning, VAR model, time dependent exposure

1 Introduction

Bayesian networks (BN) are a class of statistical models that allow an intuitive representation of the causal relationships using acyclic graphs. They are widely used in different fields such as medicine, meteorology or finance [1–3]. The learning of a BN, the process consists of two parts: a) to learn the structure of the graph and b) to learn the parameters but in this paper we will focus on structure learning. There are mostly two approaches of learning information and modelling it into a graph. The first way is to use expert's knowledge to learn the causal-effect dependencies of the field to draw a graph. It is efficient when dealing with small number of variables, but could be unfeasible for a larger number of variables. Thus, some learning methods have been proposed to learn the BN structure based from data [4,5]. These methods generally do not integrate expert's knowledge for learning the structure of BN. In order to improve these learning algorithm, a priori information such as expert's opinions has been added [6–9]. The use of a priori information has been mainly proposed in *score-based* methods in the case of time-fixed covariates in low-dimensional settings but structural restrictions in *constraint-based* algorithms have been shown as useful [10]. Using expert's opinions in *score-based* methods is achieved through an a priori distribution that makes the constraints soft; meaning that the opinions will influence the learning process but never against the data. We distinguish here opinions given with a probability $0 < p < 1$ (soft constraint) and opinions with probability $p=0$ or $p=1$ (hard constraint).

In observational settings, where most of the markers are measured repeatedly, we have shown that PC-algorithm leads in general to some edges wrongly directed from future to past in the final Bayesian network. Therefore we extended the PC-algorithm for taking into account time-varying covariates with the Chronologically order PC (COPC)-algorithm [11].

When modelling repeated measurement of covariate, the effects (edges) can be considered as (A) constant over time or (B) varying over time. These two assumptions lead to model

Bayesian network in two different ways. To model the stronger assumption (A), a new class of Bayesian network is needed to model a constant pattern which is the dynamic Bayesian network (DBN)[12]; whereas the later assumption is modelled by a classical Bayesian network. The more flexible assumption (B) is modelled by a *static* BN because there are no patterns over time. The constant over time assumption (A) can be seen as nested in the varying over time assumption (B). Structure learning methods have been largely developed for DBN [13–19], but none have added a priori expert's knowledge between different variables in this learning context.

Since anticancer immunotherapies have been developed, a key question is to identify patients who benefit most of these treatments. Biologic and immunologic values measured at treatment initiation and during the treatment could be biomarkers associated with patient's outcome including predictive biomarkers and/or early biomarkers of treatment effect. These biomarkers are collected through observational study since experimental designs in which the level of a biomarker could be set do not exist. In this context where the number of strongly correlated variables is large (i.e. such as cells of the immune system); if a biomarker has an effect on another, the learning algorithm could associate them wrongly. Therefore we choose to use a priori expert knowledge as hard constraints to reduce the searching space and the number of tests to perform.

The objective of the paper is to integrate expert's opinion as a hard constraint in the setting of time-dependent exposures case in order to improve the learning process. To achieve this objective, the COPC-algorithm initially based on the static assumption (B) will be redeveloped under the dynamic assumption (A). The skeleton of the article is as follows: in Section 2 we review some related work, then we introduce our new method to integrate experts' opinions in section 3. In section 4 we introduce the experimental settings while in section 5 we present our results from simulations and from the application to a longitudinal data set of tumour biomarkers in early breast cancer, and provide a discussion in section 6.

2 Related work

2.1 Bayesian networks definitions and notations

Bayesian networks are a class of graphical models that allows a straightforward representation of the probabilistic structure of data using graphs. Formally, based on [20] a BN is defined by $B = (G, \boldsymbol{X}, P)$, where $G = (\boldsymbol{N}, \boldsymbol{E})$ is a DAG (Directed Acyclic Graph) consisting of nodes $\boldsymbol{N} = \{N_1, \dots, N_v\}$ and edges $\boldsymbol{E}$. $\boldsymbol{X}$ represents the set of random variables $\boldsymbol{X} = \{X_1, \dots, X_v\}$ with P being the joint distribution over $\boldsymbol{X}$. Each variable of $\boldsymbol{X}$ corresponds to a single node in $\boldsymbol{N}$ and the edges represent the probabilistic dependencies between nodes. In a DAG, edges can only be directed as $X_i \rightarrow X_j$ or $X_i \leftarrow X_j$ (in the first case, X_i is a parent of X_j and X_j is a descendant of X_i). When assuming this one to one correspondence and according to the *chain rule*, the joint distribution P can be written as follow:

$$P(X_1, \dots, X_v) = \prod_1^v P(X_i \,|pa(X_i, G)), \tag{1}$$

where $pa(X_i, G)$ represents the set of parents of X_i in the DAG G (the set of variables pointing to X_i), and $P\big(X_i \,\big|pa(X_i, G)\big)$ the conditional probability of X_i giving its parents. The DAG can be seen as the map of dependencies described in the joint distributions. In fact, the DAG encodes (conditional) independence relationships through the concept of *d-separation* [21].

Somehow it may happen that several DAGs encode the same set of conditional independencies whereas the set of edges V is different. The three DAGs in (2) have the same set of conditional

independencies but are represented differently; we say that they are *Markov equivalent* or that they belong to the same *Markov equivalence class.*

$$(2) \qquad \begin{cases} X_i \rightarrow X_j \rightarrow X_k \\ X_i \leftarrow X_j \leftarrow X_k \\ X_i \leftarrow X_j \rightarrow X_k \end{cases}$$

This class encodes DAGs that have the same skeleton and *v-structures* [22]; where the skeleton is the graph we obtained by removing all arrowheads from the DAG and the *v-structures* are a triple (X_i, X_j, X_k) filling two conditions: 1) X_j is a *collider* (when two arrowheads are directed to the same node) and 2) where X_i and X_k are not adjacent. Edges which are directed differently across the DAGs in the equivalence class are represented with undirected arrows (or simply edges). These graphs representing a *Markov equivalence class* with both undirected and directed edges are called *Completed Partially DAGs* (CPDAGs) [23] or Essential graphs [24]. In the rest of this article, we will note the probability of having an arrow between X_i and X_j nodes by $p(X_i \rightarrow X_j)$, and the probability of not having an arrow such as X_i and X_j are independent $p(X_i \emptyset X_j)$ with $p(X_i \rightarrow X_j) + p(X_i \emptyset X_j) = 1$.

2.2 Learning methods

Different ways exist to learn a BN structure. A priori knowledge is generally enough for learning small DAGs, but to learn the structure of high-dimensional data, the expert's knowledge may not be sufficient. Thus, some structure learning algorithms have been developed, divided into three types: *constraint-based, score-based* and *hybrid* algorithms reviewed by Drton, Maathuis and Daly [25–27]. One of the main differences between these methods is that the *constraint-based* ones result in a PDAG while the *score-based* ones result in a DAG. Basically *constraint-based* algorithms such as PC-algorithm [4] use statistical tests to learn the skeleton of the underlying CPDAG and then use the conditional independencies found during the skeleton learning as constraint to find the v-structures and then to direct as many edges as possible according to some orientation rules. *Score-based* algorithms as described by

Heckerman [5] learn the DAG by maximizing a score for each candidate. It will select at each step the best BN candidate that maximizes the score (fitness) among all feasible edges additions, removals or reversals. The algorithm stops when the score cannot be maximized anymore. Finally the *hybrid* algorithms such as the algorithm MMHC (Max-Min Hill-Climbing) developed by Tsamardinos et al [28] combine both type of methods to learn the estimated DAG.

2.2.1 PC-algorithm

The PC-algorithm and its extensions belongs to the group of *constraint-based*. The main steps of the algorithm are (1) identification of the skeleton, (2) identification of the *v-structures* and (3) orientation of as many of the remaining edges as possible.

First, the skeleton of the underlying structure is estimated by checking all given conditional dependencies between each variable at a significance level α. Then, once the skeleton is estimated, edges are oriented in the *v-structures* to meet the conditional dependencies and finally the CPDAG is obtained by directing as many remaining edges as possible according to three rules [29]:

R1: When there is a triple $X_i \rightarrow X_j - X_k$ with X_i and X_k independent, orient $X_j - X_k$ as $X_j \rightarrow X_k$;

R2: When there is a chain $X_i \rightarrow X_k \rightarrow X_j$, $X_i - X_j$ is oriented into $X_i \rightarrow X_j$;

R3: When there are two chains $X_i - X_l \rightarrow X_j$ and $X_i - X_k \rightarrow X_j$, $X_i - X_j$ is oriented into $X_i \rightarrow X_j$ if X_k and X_l are not adjacent.

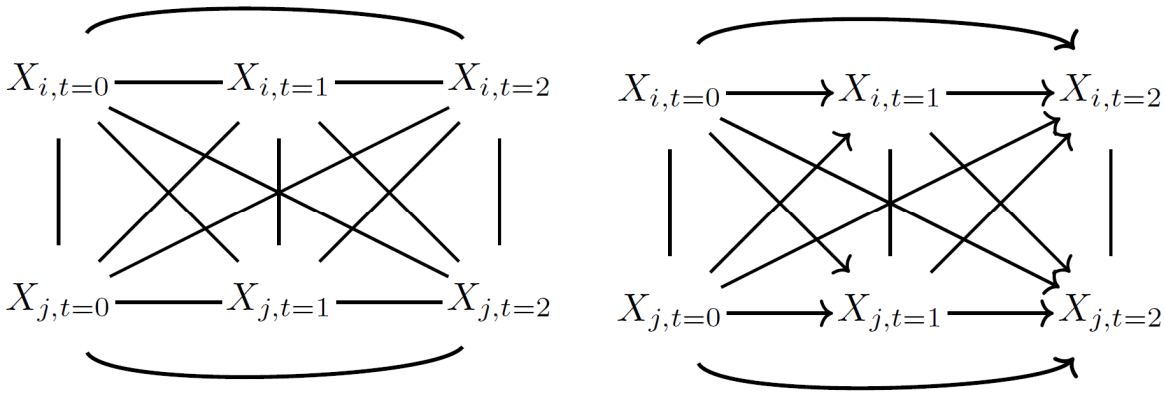
The PC-algorithm has been shown to be consistent in high-dimensional settings [30] , but poorly robust since results are impacted by the variable ordering. Two approaches have been

suggested to cope with the order dependence implemented in the step 1 of the PC-algorithm: the conservative PC-algorithm and the PC-stable [31,32]. The PC-stable is implemented in R software in the pcalg package [33].

2.2.2 The chronological order PC-algorithm

Structure learning algorithms have been developed to learn the structure of time-fixed covariates. As we have shown previously [11], when applying PC-algorithm to time-dependent exposures, the estimated CPDAG could have wrongly directed edges in terms of temporality. Therefore we have proposed the Chronological Order PC-algorithm (COPC-algorithm) which is based on the PC-stable to cope with time-dependent exposures exploiting temporal constraint in discrete time setting. Using a detailed simulation study, we have shown in the context of repeated measurements that CPDAGs estimated with COPC-algorithm were closer to the "true" CPDAG than CPDAGs using PC-stable.

We made the hypothesis that based on chronologically ordered data, the resulting CPDAG should not contain an arrow from a descendant to a parent such as $X_{1,t} \rightarrow X_{1,t'}$ where $t > t'$ since the future cannot influence a past value of the same variable. This means also that in the first step, when looking at conditional dependencies between two variables measured at time t and t' where $t \geq t'$, variables measured at a time $t^*$where $t^* \geq t$ and $t^* \geq t'$ should not be tested for the separation set **S**. We solved this issue by (1) chronologically ordering the variables in addition to the conditional independence information as input of the PC-stable algorithm as shown in figure 1 and by (2) restricting the testing of conditional independencies of $X_{i,t}, X_{j,t'} | X_{k,t^*}$ with $t^* \leq$ t and $t^* \leq t'$. In figure 1 (a) shows the initial graph without integrating repeated measures (only edges) and (b) the initial graph with a priori information on repeated measures.



(a) Initial graph without integrating repeated measures **(b) Initial graph integrating repeated measures**

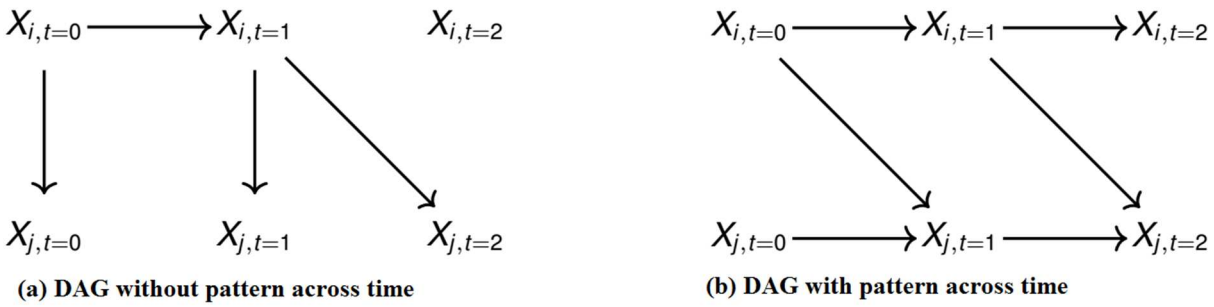
Figure 1: Initial graphs used as input without (a) and with (b) chronological a priori information for 2 variables X_i, X_j measured at 3 time points t_1, t_2 and t_3.

3 Material and methods

In this section we will first explain how we restricted the COPC-algorithm to DBN and detailed how we incorporated the expert's knowledge into the learning process of DBN. Then We will explain our simulations and application set-up.

3.1 COPC-algorithm and vector autoregressive (VAR) model

Our initial COPC-algorithm makes no assumptions about the "true" DAG as shown in figure 2a, where no pattern is present across time. When dealing with repeated measures one could assume that there exists a pattern across time such as in Figure 2b. This assumption refers to the vector autoregressive model (VAR) used to model time series with DAGs [34].



(a) DAG without pattern across time **(b) DAG with pattern across time**

Figure 2: Illustration of two possible assumptions about the true DAG with repeated measures.

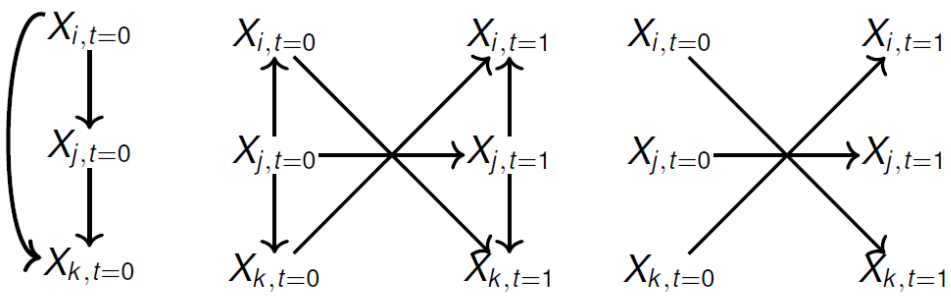
The VAR model describes the evolution of p variables across time as a linear function of their present and past values. We denote l as the time lag of the model and the representation of a VAR (l) as:

(3) $$Y_t = d + A_1 Y_{t-1} + A_2 Y_{t-2} + \cdots + A_l Y_{t-l} + \varepsilon_t,$$

where $Y_t = \{y_{1,t}, y_{2,t}, \dots, y_{p,t}\}$ is the vector of observation at time t, d is a vector of p constants, A_l is the $v * v$ matrix of coefficients of the lag l and $\varepsilon_t = \{\varepsilon_{1,t}, \varepsilon_{2,t}, \dots, \varepsilon_{p,t}\}$ is the vector of error terms. Specifying a VAR (l) model means that only variables measured up to l are in the linear function. If we consider a model VAR(2), then the linear function can be described as

(4) $$Y_t = d + A_1 Y_{t-1} + A_2 Y_{t-2} + \varepsilon_t,$$

with $A_1$the coefficients of the inter-slice edges with a lag of 1 and A_2 the coefficients of the



(a) Initial model **(b) Transition model** **(c) Transition model without intra time-slice edges**

inter-slice edges with a lag of 2.

The class of DAGs modelling the *vector autoregressive* (VAR) assumption has been called *Dynamic Bayesian Network* (DBN). They are represented by two components which are the *initial* and *transition* model illustrated in Figure 3. The *initial* model (figure 3a) represents the DAG at $t = 0$ represented by the initial joint distribution $f(X[t = 0])$. The *transition* model represents the DAG at time slices *t* to *t+l* (i.e. A_1 for the *transition* model between *t* to *t+1*). It can integrates either intra time-slice edges (figure 3b) or not (figure 3c). We will assume in the following paragraphs a VAR model with a time lag $l = 1$ with no intra time-slice edges.

Before achieving the main objective of the paper namely including a priori expert knowledge into the learning process, we have to allow the COPC-algorithm to handle the discovery of such DBN. Therefore we added a final step after having oriented the last edges. This step consists on identifying a repeated pattern inside the output graph and generalizing it to the whole graph (sketch of the added step is given in algorithm 2). Given T the number of visits of the study (usually between 3 and 6), let $T - 1$ be the total number of possible edges between $X_{i,t}$ and $X_{j,t+1}$ and F the number of observed edges between $X_{i,t}$ and $X_{j,t+1}$. Basically, for each pair of variables(X_i, X_j) in the resulting DAG, if there are more edges between F than the half of $(T - 1)$, then the algorithm forces the presence of an arrow between $X_{i,t} \rightarrow X_{j,t+1}$ at each time slice, otherwise it deletes all existing arrows between $X_{i,t}$ and $X_{j,t+1}$ at each time slice. In other words for each pair (X_i, X_j), if the pattern with a relation $(X_{i,t} \rightarrow X_{j,t+1})$ is more present than the pattern without an arrow $(X_{i,t} \emptyset X_{j,t+1})$ than it will be extend on the whole graph and vice versa.

Algorithm 2: Sketch of the added final step of DynCOPC-algorithm for VAR models

Input: DAG $\widehat{\boldsymbol{G}}$

1: For every pair $(\boldsymbol{X_i}, \boldsymbol{X_j})$ in the dag $\widehat{\boldsymbol{G}}$ **do**

2: let $\boldsymbol{F}$ be the total number of arrows between $\boldsymbol{X_{i,t}}$ and $\boldsymbol{X_{j,t+1}}$

3: If $\boldsymbol{F} \geq ((\boldsymbol{T}-\mathbf{1})/\mathbf{2})$ **then for** every time slice a directed edge is forced between $\boldsymbol{X_{i,t}}$ and $\boldsymbol{X_{j,t+1}}$ such as $\boldsymbol{X_{i,t}} \rightarrow \boldsymbol{X_{j,t+1}}$ **else**

4: for every time slice an absence of an edge is forced between $\boldsymbol{X_{i,t}}$ and $\boldsymbol{X_{j,t+1}}$ such as $\boldsymbol{X_{i,t}} \not\emptyset \boldsymbol{X_{j,t+1}}$

Output: DBN $\widehat{\boldsymbol{G}}$

When assuming the "true" DAG follows a VAR model with no intra time slice edges implies that the outputted graph is not a CPDAG but a DAG. Indeed, assuming no intra time slice edges means that the only possible edges are between two time slices (i.e. $X_{j,t} \rightarrow X_{i,t+1}$ as in Figure 3c). Therefore, undirected edges from the learned skeleton will be oriented according to the temporal constraint, leading to a final graph without undirected edges. Due to this additional step the first objective is achieved and the result is a new algorithm that we referred as *Dynamic* COPC (DynCOPC).

3.2 Integrating the expert's knowledge

3.2.1 Expert's opinions notation

In this study we assumed a VAR model with no edges between variables measured at the same time *t*. This led to have only directed edges in the *transition* model such as $X_{i,t} \rightarrow X_{j,t+1}$. A fixed ordering over the nodes has been also assumed; and so a fixed ordering over the pairs (i.e. for a pair (X_i, X_j) the first pair is $(X_{i,t=1}, X_{j,t=2})$, the second is $(X_{i,t=2}, X_{j,t=3})$, etc).In a single time point graph with n nodes, the number of pairs is given by $N = \frac{n(n-1)}{2}$. When considering two time points, having $X_{i,t} \rightarrow X_{j,t+1}$ differs from $X_{j,t} \rightarrow X_{i,t+1}$, the total number of pairs is

given by $N = n(n-1)$ and the set of total ordered pairs is noted $\boldsymbol{P}$. In our framework we assumed we had several numbers of experts (noted R) that could give a priori information on all or a small part of pairs.

Each expert k gives an opinion on the pair q that he knows as a probability of having a connection $p_{\rightarrow}^{kq} = p(X_{i,t} \rightarrow X_{j,t+1})$ or not $p_{\emptyset}^{kq} = p(X_{i,t} \emptyset\, X_{j,t+1})$. For each pair, if $p(X_{i,t} \rightarrow X_{j,t+1})$ is given by the expert, then $p(X_{i,t} \emptyset\, X_{j,t+1}) = 1 - p(X_{i,t} \rightarrow X_{j,t+1})$. Afterwards the opinions of each pair for all experts are summarized in the set $\boldsymbol{O}$ of dimension $R \times N$ such as $\boldsymbol{O} = \{o_1^1, \dots, o_q^k, \dots o_N^R\}$, where o_q^k represents the opinion for the qth pair for the kth expert with $o_q^k = \{p_{\rightarrow}^{kq}, p_{\emptyset}^{kq}\}$ if an opinion is provided and $o_q^k = \emptyset$ if not, with $p_{\rightarrow}^{kq} + p_{\emptyset}^{kq} = 1$.

Finally, opinions from all experts are merged into another set by computing the median of all experts' opinions $p_{\rightarrow}^{kq}$ and $p_{\emptyset}^{kq}$ with $p_{\rightarrow}^{kq}$ and $p_{\emptyset}^{kq}$ not empty. The new created set is then defined as $\boldsymbol{I} = \{i_1, \dots, i_N\}$ where $i_l = \{p_{\rightarrow}^{q} = median(p_{\rightarrow}^{kq}), p_{\emptyset}^{q} = median(p_{\emptyset}^{kq})\}$ and $i_k = \emptyset$ if none opinions were provided for the pair *k*.

3.2.2 Integrating expert's opinion in the COPC-algorithm

We propose to use the expert's opinion as hard constraints in the *transition* model of the DBN. We will look at each element of $\boldsymbol{I}$ and create the set $\boldsymbol{C}$ of constraints using algorithm 3, that we will use to learn the skeleton. The set of opinions $\boldsymbol{I}$ includes the merged opinions from all experts for each pair. No probabilities are assigned to a pair if no expert provides a priori information. For each pair that has a priori information, we take for the pair's maximum probability (either $P_{\rightarrow}$ or $P_{\emptyset}$) the results of a Bernoulli's process [35] and then use this for the constraint.

Algorithm 3: Sketch of the conversion from the set of experts' opinions $\boldsymbol{I}$ to the set of opinions' constraints $\boldsymbol{C}$

Input: Set of experts' opinions $\boldsymbol{I}$

1: **For** every pair q in $\boldsymbol{I}$ ***do***

2: **If** $\boldsymbol{i_q} \neq \emptyset$

3: **If** $\max(\boldsymbol{i_q}) = \boldsymbol{p_{\rightarrow}}$ **then**

4: **If** $\boldsymbol{Bernoulli}\ (\boldsymbol{p_{\rightarrow}}) = \mathbf{1}$ **then** $\boldsymbol{c_l} = \{\boldsymbol{p_{\rightarrow}} = \mathbf{1}, \boldsymbol{p_{\emptyset}} = \mathbf{0}\}$

5: **Else** $\boldsymbol{c_l} = \{\boldsymbol{p_{\rightarrow}} = \mathbf{0}, \boldsymbol{p_{\emptyset}} = \mathbf{1}\}$

6: **Else If** $\boldsymbol{Bernoulli}(\boldsymbol{p_{\emptyset}}) = \mathbf{1}$ **then** $\boldsymbol{c_l} = \{\boldsymbol{p_{\rightarrow}} = \mathbf{0}, \boldsymbol{p_{\emptyset}} = \mathbf{1}\}$

7: **Else** $\boldsymbol{c_l} = \{\boldsymbol{p_{\rightarrow}} = \mathbf{1}, \boldsymbol{p_{\emptyset}} = \mathbf{0}\}$

8: $\boldsymbol{c_l} = \emptyset$

Output: $\boldsymbol{C}$

The set $\boldsymbol{C}$ represents the constraints derived from experts' opinions that can be described as $\boldsymbol{C} = \{c_1, \dots c_N\}$ where $c_l = \{p_{\rightarrow} = 1, p_{\emptyset} = 0\}$ if the opinions led to force the presence of the edge, $c_l = \{p_{\rightarrow} = 0, p_{\emptyset} = 1\}$ if the opinions led to remove the edge and $c_k = \emptyset$ if any opinions were provided.

Once we compute the set $\boldsymbol{C}$, we can easily restrict the algorithm to learn the skeleton using algorithm 4 which is a modification of the step 1 of the COPC-stable itself derived from PC-stable [32] . In other words, we have a set of probabilities that represent the median a priori expert knowledge for each pair. For each of these probabilities we will run Bernoulli's process on the maximum probability of the pair i_q (either $p_{\rightarrow}$ or $p_{\emptyset}$). Then depending of the result of Bernoulli's' process (1 or 0) we force the presence of the arrow or not.

The algorithm 4 is the modified first step of the PC-algorithm. Originally, this step consists on creating a complete undirected graph and then testing the independence between each pair of variables according to a threshold α. At the end of the original step, the skeleton is estimated. We modified the first step as detailed in **example 1** by removing from the undirected graph the edges based on the set of constraints **C**. This means that for every pairs with $c_l = \{p_{\rightarrow} = 0, p_{\emptyset} = 1\}$ we remove it from the graph G and so this pair will not be tested all along the algorithm**.** The step 7 is modified in restricting the search over the pairs in $\boldsymbol{C}$ that did not have

any opinions ($c_k = \emptyset$). Pairs in $\boldsymbol{C}$ that are independent are removed from the search at step 2 and remaining pairs with $c_k = \{p_{\rightarrow} = 1, p_{\emptyset} = 0\}$ will not be tested. This step leads that pairs with a probability of $p_{\rightarrow} = 1$ to be in the final graph. We refer to these methods as $\mathrm{COPC}_{\mathrm{expert}}$ (COPC using algorithms 3 and 4) and dynamic $\mathrm{COPC}_{\mathrm{expert}}$ $(\mathrm{DynCOPC}_{\mathrm{expert}})$when using algorithms 2, 3 and 4 with COPC.

Algorithm 4: Sketch of the first step of the COPC-algorithm to integrate a priori expert's knowledge

---

Input: The set of ordered pairs $\boldsymbol{P}$, significance parameter α, the set of expert's constraint $\boldsymbol{C}$

1: Form an undirected graph G that respects **a** time lag of 1 with no edges between variables of a same time t as in Figure 4b

2: **For** every pair $\boldsymbol{q}$ in $\boldsymbol{C}$ **do**

If $\boldsymbol{c_l} = \{\boldsymbol{p_{\rightarrow}} = \mathbf{0}, \boldsymbol{p_{\emptyset}} = \mathbf{1}\}$ **then** remove the edge in G for the $\boldsymbol{q^{th}}$ pair of $\boldsymbol{C}$ such as $\boldsymbol{X_{i,t}}\ \emptyset\ \boldsymbol{X_{j,t+1}}$ at each time lag

end

3: $\boldsymbol{b} = -\mathbf{1}$

4: **Repeat**

5: $\boldsymbol{b} = \boldsymbol{b} + \mathbf{1}$

6: **Repeat**

7: Using $\boldsymbol{P}$, select a (new) pair q $(\boldsymbol{X_{i,t}}, \boldsymbol{X_{j,t+1}})$ that is adjacent in G satisfying $\left|\boldsymbol{adj}(\boldsymbol{X_{i,t}})\backslash \boldsymbol{X_{j,t+1}}\right| > \boldsymbol{b}$ and $\boldsymbol{c_l} = \emptyset$

8: **Repeat**

9: Choose a (new) set $\boldsymbol{S} \subseteq \boldsymbol{adj}(\boldsymbol{X_{i,t}})\backslash \{\boldsymbol{X_{j,t+1}}, \boldsymbol{X_{k,t'}}\}$ with $|\boldsymbol{S}| = \boldsymbol{b}$ and $\boldsymbol{t'} > \boldsymbol{t} + \mathbf{1} > \boldsymbol{t}$

10: If $\boldsymbol{X_{i,t}}$ and $\boldsymbol{X_{j,t+1}}$ are conditionally independent given S **then**

Remove edge $\boldsymbol{X_{i,t}} - \boldsymbol{X_{j,t+1}}$ from G

Let $\boldsymbol{S_{sep}}$ be the separation set $\boldsymbol{S_{sep}}$ $(\boldsymbol{X_{i,t}}, \boldsymbol{X_{j,t+1}}) = \boldsymbol{S}$

11: **Until** $\boldsymbol{X_{i,t}}$ **and** $\boldsymbol{X_{j,t+1}}$ are no longer adjacent or all $\subseteq \boldsymbol{adj}(\boldsymbol{X_{i,t}})\backslash \{\boldsymbol{X_{j,t+1}}, \boldsymbol{X_{k,t'}}\}$ with $|\boldsymbol{S}| = \boldsymbol{b}$ and $\boldsymbol{t'} > \boldsymbol{t} + \mathbf{1} > \boldsymbol{t}$ have been tested

12: **Until** all ordered pairs of adjacent nodes $\boldsymbol{X_{i,t}}, \boldsymbol{X_{j,t+1}}$ with $\left|\boldsymbol{adj}(\boldsymbol{X_{i,t}})\backslash \boldsymbol{X_{j,t+1}}\right| > \boldsymbol{b}$ have been tested

13: **Until** all pairs of adjacent nodes $\boldsymbol{X_{i,t}}, \boldsymbol{X_{j,t+1}}$ satisfy $\left|\boldsymbol{adj}(\boldsymbol{X_{i,t}})\backslash \boldsymbol{X_{j,t+1}}\right| \leq \boldsymbol{l}$

Output G, $\boldsymbol{S_{sep}}$

Example 1: Let Figure 3a be the "true" DAG, Figure 3b the undirected graph formed at step 1 of algorithm 4 and $\boldsymbol{C} = \{X_{i,t}, X_{i,t+1} = \emptyset;\ X_{i,t}, X_{j,t+1} = \{1,0\}; X_{i,t}, X_{k,t+1} = \emptyset; X_{j,t}, X_{j,t+1} = \emptyset; X_{j,t}, X_{i,t+1} = \{0,1\}; X_{j,t}, X_{k,t+1} = \emptyset; X_{k,t}, X_{k,t+1} = \emptyset;\ X_{k,t}, X_{i,t+1} = \{0,1\}; X_{k,t}, X_{j,t+1} = \emptyset\}$ the expert's opinion.

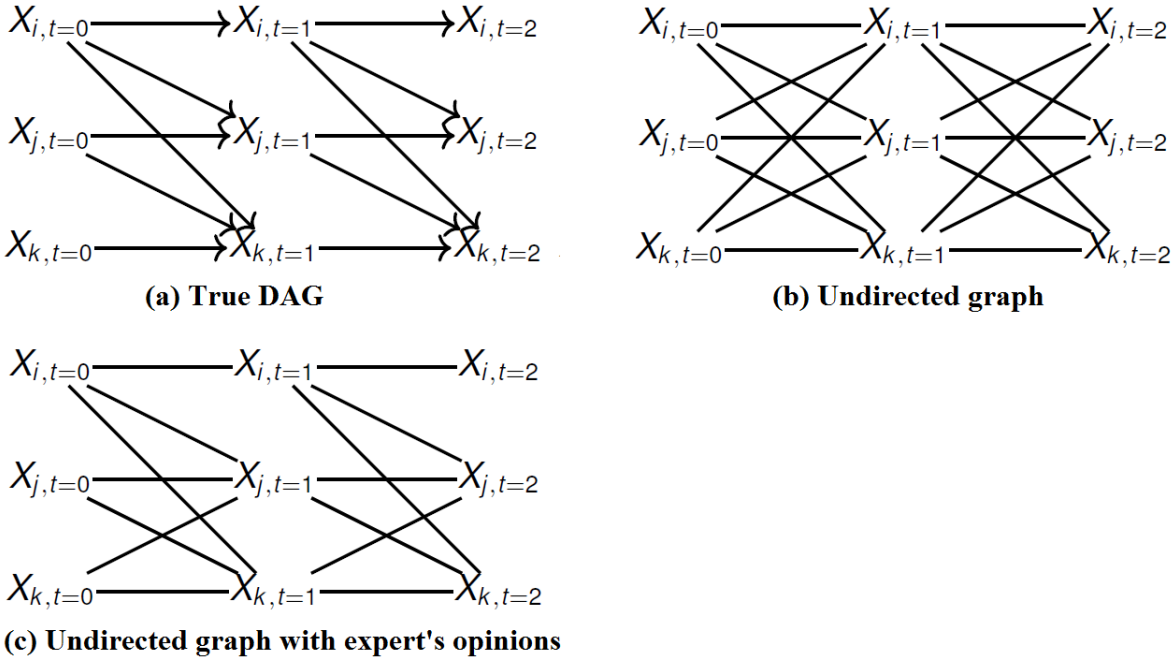


Figure 4: Illustration of the example 1 with the true DAG in (a), the undirected graph that fits a first order VAR model in (b) and the undirected graph obtained based on restrictions given by the experts.

In other words, the original step 1 of the PC-algorithm is to test every conditional independencies, but here in the step 2 we remove all hard constraints that we created using algorithm 3. Then from step 6 to 13, we test all the other pairs for conditional independencies.

4 Experimental setting

4.1 Simulations

To measure the efficiency of our new method and attest the efficiency of a priori expert's knowledge as hard constraint, we ran a bench of simulations. Based from a random "true"

DAG, we wanted to generate a dataset with different time-points (typically clinical visits in the context of clinical research) and to recover it using our algorithm with and without the use of a priori knowledge. We assumed that a group of experts with different levels will give their opinions about a percentage of the total information available. Then we compared COPC, $\mathrm{COPC}_{\mathrm{expert}}$, DynCOPC and $\mathrm{DynCOPC}_{\mathrm{expert}}$. The details are described in Appendix 1.

4.2 Application

To validate our method on real word data, we applied it to a longitudinal clinical study [39], i.e. a neoadjuvant phase II trial of letrozole in estrogen receptor-positive breast cancer patients with the expression of genes in the tumour as biomarkers. The gene expression data of this clinical study is publicly available in the gene expression omnibus (GEO). To evaluate the efficacy of our methods, we restricted the search over the cell cycle graph from KEGG [2], as this is the main biological pathway in estrogen-positive breast cancer, and used it as the "true" graph to compare the results. This represents a total of 44 genes measured 3 times during the study on 56 patients (baseline, 14 weeks after treatment and 3 months after treatment). We also used the original graph from KEGG to simulate expert knowledge. Due to a high number of possible pairs, we only used 5% of priory expert knowledge to be realistic (0.05x44x43=94).

5 Results

5.1 Simulations

The results of the simulations are presented in table 1. Since the differences of the standardized Hamming distance did not vary for the significance parameter α, we only presented results for α= 0.05. The amount of experts' opinions varies from 5% to 40% of the total percentage of possible pairs of variables; which corresponds to provide 20 to 160 pairs of variables by the experts.

The results of Hamming distance showed that DynCOPC obtained lower Hamming distance values compared to COPC and PC-stable with and without expert's opinions despite all other scenarios. Also the Hamming distance among all algorithms increased with the number of time-points.

For *perfect* experts, adding 5% or 40% a priori knowledge resulted in reducing the Hamming distance in COPC and DynCOPC algorithms. The reduction of Hamming distance was more substantial when using 40% of a priori information. However we observed that incorporating 5% of *perfect* opinions in the PC-stable resulted in an augmentation of the Hamming distance.

For *bad* experts, it led to an increase of the Hamming distance. Nevertheless we noticed that even with *bad* opinions, the $\mathrm{DynCOPC}_{\mathrm{expert}}$ got better or similar results than $\mathrm{COPC}_{\mathrm{expert}}$ with *perfect* opinions when using 5% of a priori information (22, 16 versus 25 with T=4 and 34, 27 versus 34 with T=8). The PC-stable on the other hand had the worst performance with the highest Hamming distance in all scenarios.

More detailed results on different scenarios are available in appendix 1.

Table 1: Average standardized Hamming distance according to the different algorithms over 500 random DAGs with 20 nodes with α=0.05. Bold and underlined values report standardized Hamming distance smaller than COPC or DynCOPC respectively.

T	**Expert's opinion**	**Level**	**PC-stable$_{exp}$ (sd)**	**COPC_{exp} Algo (sd)**	**$\mathrm{DynCOPC}_{exp}$ Algo (sd)**	**PC-stable (sd)**	**COPC Algo (sd)**	**DynCOPC Algo (sd)**
4	5%	*bad*	45 (2)	37 (4)	**22** (4)	32 (2)	26 (4)	17 (3)
		perfect	34 (3)	**25** (4)	**<u>16</u>** (4)			
	40%	*bad*	110 (4)	107 (5)	56 (4)			
		perfect	**22** (4)	**<u>16</u>** (3)	**<u>11</u>** (2)			
8	5%	*bad*	52 (3	47 (4)	**34** (3)	38 (3)	35 (4)	28 (3)
		perfect	40 (2)	**34** (4)	**<u>27</u>** (3)			
	40%	*bad*	127 (5)	127 (6)	71 (5)			

perfect	**26** (3)	**22** (3)	**19** (3)

5.2 Application

The results of the application are given in table 2. Globally all algorithms with expert's knowledge had a better recovery than without. As expected, the $\mathrm{DynCOPC_{exp}}$ outperformed all the other algorithms with the lowest SHD (122). We observed that $\mathrm{PC_{exp}}$ had a lower SHD than PC but had a similar SHD with DynCOPC (≈ 130). Surprisingly COPC had a higher SHD than PC.

Table 2: Average standardized Hamming distance according to the different algorithms over 500 random datasets and set of constraints (5%). The significance level was set at α=0.05.

PC (sd)	**COPC Algo (sd)**	**DynCOPC Algo (sd)**	**$\mathrm{PC_{exp}}$ (sd)**	**$\mathrm{COPC_{exp}}$ Algo (sd)**	**$\mathrm{DynCOPC_{exp}}$ Algo (sd)**
136 (3)	140 (4)	130 (3)	131 (3)	126 (3)	122 (3)

6 Discussion

In this article we proposed a method to integrate expert's opinion in causal learning methods such as *constraint-based* algorithms in the case of repeated and multi-dimensional settings which has not been done until now, to our knowledge. The framework is built by translating

expert's opinion in constraints used in the algorithm. Giving multiple experts' opinions and their uncertainties, it will return a set of constraints that will be used in the algorithm. These constraints could either force the presence of an edge or force the removal of it. In fact we proposed here two algorithms in the case of repeated and multi-dimensional settings, one that follows the VAR assumption with a single pattern across time (DynCOPC) and one that does not (COPC). Since the VAR assumption corresponds to a specific case for repeated measures (where the structure is constant over time), we have based our simulations on this. The COPC can then recover a more general pattern (i.e, where the structure cannot be constant over time) in repeated measures while DynCOPC is defined only for VAR assumption.

A priori experts' information in structure learning methods has mainly be used but in low-dimensional setting [6,36,37]. In the case of large number of variables, it is difficult to assume that all a priori information is known and that only a certain percentage of opinions can be given. Therefore we explored the incorporation of a priori information from 5% to 40%. We also ran sensitivity analysis where we tested the effect of modifying the set of constraints over several datasets and the results of one set of constraints over several datasets. These modifications did not impact the main results and DynCOPC had still a better recovery than COPC and PC.

We have shown that when using only 5% of total information as a priori constraints, it was possible to reduce the Hamming distance for *perfect* experts compared to both versions of algorithms without a priori information. The reduction of the Hamming distance was more important with the percentage of a priori information. However in real situations, we expect that it is more realistic to assume that expert will give a small percentage of opinions, close to 5% of total a priori information rather than 40% or more. In fact we tested our method on a real dataset from a nonrandomized neoadjuvant phase II trial of letrozole in estrogen receptorpositive (ERþ) breast cancer patients with genes as biomarkers. The results showed that the SHD is reduced by the integrating of expert's opinions as constraints leading to improved

graph learning. Contrary to COPC and PC, the DynCOPC is specially designed to learn Dynamic DAG and thus performed better.

The ways of integrating expert's knowledge in structure learning methods has been intensively studied [29,38–43]. Meek proposed to integrate the expert knowledge after the learning process to complete it and helping to orient undirected edges. In the case where the expert knowledge would contradict the results obtained from the data, the results obtained with data prevail over the expert knowledge. So in this implementation, the expert knowledge is not used a priori but a posteriori. Tan proposed a method to integrate expert knowledge as a modification of the significance parameter α in the PC-algorithm [40]. However the calculation was based on a "trust" parameter of the given information that is difficult to estimate in a real situation. Richardson focused on how to merge properly opinions from different experts in the case of *score-based* methods [41]. Of note, several opinions on a same pair of variables can be contradictory. In our work we calculated the median of all probabilities but it is possible in a further work for example to weight experts' opinions according to expert confidence in their opinion.

Recently Amirkhani added some errors in the given opinions in *score-based* method by simulating the bad, mediocre and good experts [6]. Each expert regardless of his level had the same probability to give wrong or correct information about a pair of variables; while in our method we simulated different experts that give information about a pair of variables according to a range of probabilities..

The assumptions of our methods are well defined. We supposed to have Gaussian covariates measured at a discrete time interval. Also to simplify the incorporation of expert's opinions, we made the assumption that the true DAG had a unique pattern across time (first order VAR model) and that expert's beliefs are also constant across time. This may be true in economics where the VAR model is widely used but medicine and biology are known to be much more

complex. For example [44] showed that the T cell activation was determined by the number of T cell receptor (TCR). They observed that T cells responded only when the number of triggered TCR reached a number of 8000, meaning that when the number of TCR was below this threshold, the T cells were inactive. This illustrates clearly that the T cells activations and responses are varying within time and that a dynamic Bayesian network may not be suitable to model this kind of pathway.

We also considered that there were no edges between variables measured at the same time t. The interpretation of such an edge differs from an edge between time t and t' (with $t > t'$) . If we refer to Allen's theory [45], edges between variables measured at a same time t can only equal, overlap or meet each other; so the establishment of the dependence between $X_{i,t}$ and $X_{j,t}$ is much more complex than just a test. Therefore in this study we focused on relations where X_i occurs strictly before X_j.

Our method to convert experts' opinions as constraints is given in algorithm 3. Using the Bernoulli's process makes it nondeterministic and thus, repeating it on a same set $\boldsymbol{I}$ will give different sets of constraints $\boldsymbol{C}$. We choose to focus on this kind of approaches rather than deterministic ones because restricting the opinions probabilities to a single threshold reduces the variability of a given opinion. In such approaches, where different models can be obtained from the changes of the set $\boldsymbol{C}$; further work can be done by developing methods that combine them and get a better graph. In parallel, learning Bayesian network was not only for learning the dependencies between variables but also for estimating causal effects.

Most of the methods integrating a priori information check whether the opinions fit the data, and reject them if not. Our method does not check whether the expert opinion is substantiated by the data or contradicts the data. In a domain where the biomarkers are strongly correlated, if a biomarker has an effect on another, the algorithm could often associate them wrongly and vice versa. Using hard constraints is the only way to avoid false positive relations under the assumption that the information is correct. This implies that we collaborate with high level

experts. Being a *bad* expert does not mean that the expert does not know, but it is rather an expert that will put a strong probability on a false relation. In the context of immuno-oncology, having *bad* experts or wrong a priori information is unlikely since the experts are generally highly renowned in their fields and their knowledge is derived from in-vitro experiments, fundamental knowledge about immunology where the level of evidence is strong.

In addition, one could note that the hard constraints are inherent to the field of immunology. Only known immunological biomarkers are measured and biomarkers which have a potential causal effect according to previous knowledge are much more susceptible to be analysed. Since this selection of immunological biomarkers of interest is inherent to the field and can be seen as hard constraint, it is somewhat logical to integrate a priori knowledge as hard constraints.

In this study we choose to use expert's opinions as a priori information but it exists other sources than can be used as a priori information. In immunology, experimental studies such as in vitro studies are available, or machine learning methods allow to learn relations by searching into the bibliography. Integrate such information in our method can be done by considering it directly in the set of constraints $\boldsymbol{C}$.

Non-experimental studies are of interest for identifying candidate biomarkers from large number of measured variables. Of note, these variables could be highly correlated as are the immunological markers of the immune system cells. In previous work, biomarkers have been identified in the context of observational studies with a large number of variables using causal inference [46]. The principle is to learn a CPDAG (Completed partially DAG) and then estimated causal effects based on the estimated graph using *do-calculus* [47]. In the context of immuno-oncology, a priori information is available based on renowned experts or experimental results to improve the learning of the CPDAG, and so the estimated causal effects. But when data follows VAR assumption, the classical definition of causality does not apply and the Granger causality has to be applied [48,49].

Conclusions

We presented in this paper an original and efficient method to integrate a priori expert knowledge in *constraint-based* structure learning algorithms on repeated measures. In fact, it allows taking into account opinions from several experts in the context of repeated measures and then converting them into hard constraints that modified the output of the final graph. It led to a reduced searching space and a better recovery of the true structure in terms of Hamming distance with all versions of algorithms PC, COPC and DynCOPC when using at least 5% of total information as a priori. Our method is new since the constraints are not checked with the data for letting opinions contradict the data. This is pertinent when the data are strongly correlated: only the hard constraints can be used to recover the true structure since false positive association can arise from the data in lack of these hard constraints.

In a further work we want to apply this method to immunological data after having integrated a range of highly renowned experts and a priori information based on experimental studies as input and implementing the Granger causality to estimate causal effects in the context of longitudinal data and identify biomarkers that are associated with treatment response or toxicity of immunotherapies. Other future works that can be done such as the improvement of merging expert's opinions, the merging of different graphs obtained by varying $\boldsymbol{C}$.

Acknowledgements

This work is part of a PhD thesis funded by Université Paris Sud.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

VA, PL, SM, EM have conceived the statistical work, VA has drafted the manuscript. All authors have critically reviewed the manuscript.

References

[1] Petousis P, Han SX, Aberle D, Bui AAT. Prediction of lung cancer incidence on the low-dose computed tomography arm of the National Lung Screening Trial: A dynamic Bayesian network. Artif Intell Med 2016;72:42–55. doi:10.1016/j.artmed.2016.07.001.

[2] Kocadağli O, Aşikgil B. Nonlinear time series forecasting with Bayesian neural networks. Expert Syst Appl 2014;41:6596–610. doi:10.1016/j.eswa.2014.04.035.

[3] Dabrowski JJ, Beyers C, de Villiers JP. Systemic banking crisis early warning systems using dynamic Bayesian networks. Expert Syst Appl 2016;62:225–42. doi:10.1016/j.eswa.2016.06.024.

[4] Spirtes P, Glymour C, Scheines R. Causation, Prediction and Search. 2nd ed. MIT Press; 2000.

[5] Heckerman D, Geiger D, Chickering DM. Learning Bayesian Networks: The Combination of Knowledge and Statistical Data. Mach Learn 1995;20:197–243. doi:10.1023/A:1022623210503.

[6] Amirkhani H, Rahmati M, Lucas PJF, Hommersom A. Exploiting Experts' Knowledge for Structure Learning of Bayesian Networks. IEEE Trans Pattern Anal Mach Intell 2017;39:2154–70. doi:10.1109/TPAMI.2016.2636828.

[7] Ben Messaoud M, Leray P, Ben Amor N. Integrating Ontological Knowledge for Iterative Causal Discovery and Visualization. Lect Notes Comput Sci (Including Subser Lect Notes Artif Intell Lect Notes Bioinformatics) 2009;5590 LNAI:168–79. doi:10.1007/978-3-642-02906-6_16.

[8] Masegosa AR, Moral S. An interactive approach for Bayesian network learning using domain/expert knowledge. Int J Approx Reason 2013;54:1168–81. doi:10.1016/j.ijar.2013.03.009.

[9] Sousa HS, Prieto-Castrillo F, Matos JC, Branco JM, Lourenço PB. Combination of expert decision and learned based Bayesian Networks for multi-scale mechanical analysis of timber elements. Expert Syst Appl 2018;93:156–68. doi:10.1016/j.eswa.2017.09.060.

[10] de Campos LM, Castellano JG. Bayesian network learning algorithms using structural restrictions. Int J Approx Reason 2007;45:233–54. doi:10.1016/j.ijar.2006.06.009.

[11] Asvatourian V, Coutzac C, Chaput N, Robert C, Michiels S, Lanoy E. Estimating causal effects of time-dependent exposures on a binary endpoint in a high-dimensional setting. BMC Med Res Methodol 2018:1–12. doi:10.1186/s12874-018-0527-5.

[12] Dean T, Kanazawa K. A model for reasonning about persistence and causation. Comput Intell 5(3) 1989;1:142–50.

[13] Trabelsi G, Leray P, Ben Ayed M, Alimi AM. Dynamic MMHC: A Local Search Algorithm for Dynamic Bayesian Network Structure Learning. Twelfth Int Symp Intell Data Anal 2013:392–403. doi:10.1007/978-3-642-41398-8_34.

[14] Moneta A, Chlaß N, Entner D. Causal Search in Structural Vector Autoregressive Models. Jmlr 2011;12:95–118.

[15] Chu T, Glymour C. Search for Additive Nonlinear Time Series Causal Models. J Mach Learn Res 2008;9:967–91.

[16] Opgen-Rhein R, Strimmer K. Learning causal networks from systems biology time course data: An effective model selection procedure for the vector autoregressive process. BMC Bioinformatics 2007;8:1–8. doi:10.1186/1471-2105-8-S2-S3.

[17] Dojer N, Gambin A, Mizera A, Wilczyński B, Tiuryn J. Applying dynamic Bayesian networks to

perturbed gene expression data. BMC Bioinformatics 2006;7:1–11. doi:10.1186/1471-2105-7-249.

[18] Murphy KP. Dynamic Bayesian Networks: Representation, Inference and Learning. University of California, 2002. doi:10.1016/j.dss.2003.08.004.

[19] Vinh NX, Chetty M, Coppel R, Wangikar PP. Local and Global Algorithms for Learning Dynamic Bayesian Networks. 2012 IEEE 12th Int Conf Data Min 2012:685–94. doi:10.1109/ICDM.2012.18.

[20] Scutari M, Denis JB. Bayesian Networks - With Examples in R. Chapman and Hall; 2015. doi:10.1145/1859204.1859227.

[21] Pearl J. Causal diagrams for empirical research. Biometrika 1995;82:669–88. doi:10.1093/biomet/82.4.669.

[22] Verma T, Pearl J. Equivalence and synthesis of causal models. Proc Sixth Conf Uncertain Artif Intell 1991:200–27.

[23] Chickering DM. Optimal Structure Identification With Greedy Search. J Mach Learn Res 2002;1:507–54. doi:10.1162/153244303321897717.

[24] Andersson SA, Madigan D, Perlman MD. A characterization of Markov equivalence classes for acyclic diagraphs. Ann Stat 1997;25:505–41. doi:10.1214/aos/1031833662.

[25] Drton M, Maathuis MH. Structure Learning in Graphical Modeling. Annu Rev Stat Its Appl 2016. doi:10.1146/annurev-statistics-060116-053803.

[26] Maathuis MH, Nandy P. A review of some recent advances in causal inference. Handb. big data, Chapman and Hall; 2016.

[27] Daly R, Shen Q, Aitken S. Learning Bayesian Networks: Approaches and Issues. Knowl Eng Rev 2011. doi:10.1017/S0269888910000251.

[28] Tsamardinos I, Brown LE, Aliferis CF. The max-min hill-climbing Bayesian network structure learning algorithm. Mach Learn 2006;65:31–78. doi:10.1007/s10994-006-6889-7.

[29] Meek C. Causal inference and causal explanation with background knowledge. Proceeding UAI'95 Proc Elev Conf Uncertain Artif Intell 1995:403–10.

[30] Kalisch M, Buehlmann P. Estimating high-dimensional directed acyclic graphs with the PC-algorithm. J Mach Learn Res 2005;8:613–36.

[31] Ramsey J, Zhang J, Spirtes PL. Adjacency-Faithfulness and Conservative Causal Inference. Proc Twenty-Second Conf Uncertain Artif Intell 2006:401–8.

[32] Colombo D, Maathuis MH. Order-independent constraint-based causal structure learning. J Mach Learn Res 2014;15:1–40. doi:papers3://publication/uuid/CC1A353F-4C95-4058-93CC-05855FD6E5E3.

[33] Kalisch M, Machler M, Colombo D, Maathuis MH, Buhlmann P, Mächler M, et al. Causal Inference Using Graphical Models with the R Package pcalg. J Stat Softw 2012;47:26. doi:papers3://publication/uuid/9A118F8A-B35B-4F17-B8CF-3D8B39362A4A.

[34] Moneta A. Graphical causal models and VARs: An empirical assessment of the real business cycles hypothesis. Empir Econ 2008;35:275–300. doi:10.1007/s00181-007-0159-9.

[35] McCullagh P, Nelder J. Generalized Linear Models. Chapman and Hall/CRC; 1989.

[36] Oates CJ, Kasza J, Simpson JA, Forbes AB. Repair of Partly Misspecified Causal Diagrams. Epidemiology 2017;28:548–52. doi:10.1097/EDE.0000000000000659.

[37] Castelo R, Siebes A. Priors on network structures. Biasing the search for Bayesian networks. Int

J Approx Reason 2000;24:39–57. doi:10.1016/S0888-613X(99)00041-9.

[38] Angelopoulos N, Cussens J. Bayesian learning of Bayesian networks with informative priors. Ann Math Artif Intell 2008;54:53–98. doi:10.1007/s10472-009-9133-x.

[39] Constantinou AC, Fenton N, Neil M. Integrating expert knowledge with data in Bayesian networks: Preserving data-driven expectations when the expert variables remain unobserved. Expert Syst Appl 2016;56:197–208. doi:10.1016/j.eswa.2016.02.050.

[40] Tan M, Alshalalfa M, Alhajj R, Polat F. Combining multiple types of biological data in constraint-based learning of gene regulatory networks. Comput Intell Bioinforma Comput Biol 2008 CIBCB'08 IEEE Symp 2008:90–7. doi:10.1109/CIBCB.2008.4675764.

[41] Richardson M, Domingos P. Learning with knowledge from multiple experts. Mach Learn Work Then Conf 2003;20:624.

[42] Borboudakis G, Tsamardinos I. Scoring and Searching over Bayesian Networks with Causal and Associative Priors. arXiv:14082057 [csAI] 2012.

[43] Su C, Borsuk ME, Andrew A, Karagas M. Incorporating prior expert knowledge in learning Bayesian networks from genetic epidemiological data. Comput Intell Bioinforma Comput Biol 2014 IEEE Conf 2014:1–5. doi:10.1109/CIBCB.2014.6845507.

[44] Viola A, Lanzavecchia A. T Cell Activation Determined by T Cell Receptor Number and Tunable Thresholds Author (s): Antonella Viola and Antonio Lanzavecchia Published by : American Association for the Advancement of Science Stable URL : http://www.jstor.org/stable/2890056. Adv Sci 1996;273:104–6.

[45] Allen J. Towards a general theory of action and time. Artif Intell 1984;23:123–54. doi:10.1016/0004-3702(84)90008-0.

[46] Maathuis MH, Kalisch M, Bühlmann P. Estimating high-dimensional intervention effects from observational data. Ann Stat 2009;37:3133–64. doi:10.1214/09-AOS685.

[47] Pearl J. Causality: models, reasoning, and inference. Cambridge university press; 2000.

[48] Eichler M, Didelez V. Causal Reasoning in Graphical Time Series Models. Uncertain Artif Intell 2009;1:1–8.

[49] Eichler M, Didelez V. On Granger causality and the effect of interventions in time series. Lifetime Data Anal 2010;16:3–32. doi:10.1007/s10985-009-9143-3.

List of abbreviations

DAG: Directed Acyclic Graph

CPDAG: Completed partially Directed Acyclic Graph

PC-algorithm: Peter Clarks -algorithm

COPC: Chronologically ordered PC DynCOPC: Dynamic Chronologically ordered PC

VAR: Vector Autoregressive